IN THE SPECIFICATION

Please amend the specification as follows.

Please replace the paragraph beginning on page 7 line 14 with the following rewritten paragraph.

Each upload proxy server ~~send~~ sends a message acknowledging receipt of data sent to it by a client.

Please replace the paragraph beginning on page 8 line 15 with the following rewritten paragraph.

The method also includes the step of uploading data from at least one of the upload proxy servers to the common destination server responsive to messages sent indicating that the particular upload proxy server is holding data for the common destination server. The method ~~of~~ further includes the step of separately time-stamping the unique identifiers as received by the authenticator; and wherein the step of separately sending back to each client a message, digitally signed by the authenticator, with the unique identifier sent by that client includes the corresponding time-stamp within the message; and wherein the common destination server uses the unique identifier for the data provided by each client to confirm that the data provided by each client existed as of the corresponding time-stamp and to insure that the data.

Please replace the paragraph beginning on page 15 line 15 with the following rewritten paragraph.

At block 306 of FIG. 4, the authenticator time-stamps the received message from the client with a time stamp σ. At block 308, the authenticator concatenates h(T) with σ and encrypts the result with the private key of event Kpriv. As also shown by message 128, the result ε = Kpriv (h(T), σ), an upload ticket, is sent to the client 112. It may also optionally ~~sends~~ send a list of upload proxy servers that the client may use. It should be ~~understand~~ understood that encrypting the data with the event private key is the same as digitally signing the data.

Please replace the paragraph beginning on page 15 line 22 with the following rewritten paragraph.

At block ~~310~~ 309 of FIG. 4, the client 112 inspects ε to make sure it was generated by the authenticator and not some imposter.

Please replace the paragraph beginning on page 15 line 25 with the following rewritten paragraph.

At block ~~312~~ 310 of FIG. 4, the client 112 generates a session key Kses (length set in the EID) and encrypts T with Kses according to a known symmetric-key cryptographic standard (such as DES or triple DES) specified in EID. Client 112 then concatenates Kses with ε, encrypts Kses with the public key Kpub of the event, and, at block ~~314~~ 312, sends EID, Kses(T), and Kpub(Kses, ε) to one of the upload proxy servers 116 (only one shown in FIG. 5). (The selection of a particular upload proxy server will be discussed below.) Message 130 of FIG. 5 is the data being sent to upload proxy server 116, but it will be understood that the data is encrypted in the stated manner except that the EID is sent in the clear

(i.e., unencrypted form). Therefore, the owner of the destination server 110 does not need to trust the owner of the particular upload proxy server 116 through which the data is sent. Any alteration in the data can be detected.

Please replace the paragraph beginning on page 16 line 11 with the following rewritten paragraph.

As also indicated in block ~~314~~ 312, the upload proxy server sends a receipt (not shown in FIG. 5) to the client. The receipt is preferably digitally signed by a private key used by the particular upload proxy server. The receipt may be the upload proxy server's digital signature of the EID, Kses(T), and Kpub(Kses, ε). This helps guard against so-called man-in-the-middle attacks.

Please replace the paragraph beginning on page 16 line 23 with the following rewritten paragraph.

At block 402 of FIG. 6, the upload proxy server 116 sends the receipt to the destination server 110. (This receipt is sent at about the same time as the receipt sent to the client as discussed above in connection with block ~~314~~ 312.) The receipt advises server 110 that server 116 has a submission for EID. At block 404, the server 110 records the receipt and the identity of the upload proxy server holding the information for it. After the deadline of the event (if applicable) or otherwise at a later time, and at block 406, the destination server 110 uploads the data from each of the upload proxy servers 116. This corresponds to message 132 of FIG. 5.